

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

IN RE APPLICATION OF: Koichi KONDO, et al.

GAU:

SERIAL NO: New Application

EXAMINER:

FILED: Herewith

FOR: MECHANISM SIMULATION METHOD AND MECHANISM SIMULATION PROGRAM

**REQUEST FOR PRIORITY**

COMMISSIONER FOR PATENTS  
ALEXANDRIA, VIRGINIA 22313

SIR:

- Full benefit of the filing date of U.S. Application Serial Number , filed , is claimed pursuant to the provisions of 35 U.S.C. §120.
- Full benefit of the filing date(s) of U.S. Provisional Application(s) is claimed pursuant to the provisions of 35 U.S.C. §119(e): Application No. Date Filed
- Applicants claim any right to priority from any earlier filed applications to which they may be entitled pursuant to the provisions of 35 U.S.C. §119, as noted below.

In the matter of the above-identified application for patent, notice is hereby given that the applicants claim as priority:

<u>COUNTRY</u>	<u>APPLICATION NUMBER</u>	<u>MONTH/DAY/YEAR</u>
Japan	2002-376209	December 26, 2002
Japan	2003-389710	November 19, 2003

Certified copies of the corresponding Convention Application(s)

- are submitted herewith
- will be submitted prior to payment of the Final Fee
- were filed in prior application Serial No. filed
- were submitted to the International Bureau in PCT Application Number  
Receipt of the certified copies by the International Bureau in a timely manner under PCT Rule 17.1(a) has been acknowledged as evidenced by the attached PCT/IB/304.
- (A) Application Serial No.(s) were filed in prior application Serial No. filed ; and
- (B) Application Serial No.(s)  
 are submitted herewith  
 will be submitted prior to payment of the Final Fee

Respectfully Submitted,

OBLON, SPIVAK, McCLELLAND,  
MAIER & NEUSTADT, P.C.



Marvin J. Spivak

Registration No. 24,913

C. Irvin McClelland  
Registration number 21,124

Customer Number

22850

Tel. (703) 413-3000  
Fax. (703) 413-2220  
(OSMMN 05/03)

247082USA25RD  
0380959

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 Date of Application: 2002年12月26日

出願番号 Application Number: 特願2002-376209

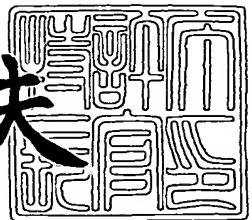
[ST. 10/C]: [JP2002-376209]

出願人 Applicant(s): 株式会社東芝

2003年 7月18日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



【書類名】 特許願  
【整理番号】 13B02Y0431  
【あて先】 特許庁長官殿  
【国際特許分類】 G06F 15/00  
【発明の名称】 機構シミュレーション方法および機構シミュレーション  
プログラム  
【請求項の数】 9  
【発明者】  
【住所又は居所】 神奈川県川崎市幸区小向東芝町1番地 株式会社東芝  
研究開発センター内  
【氏名】 近藤 浩一  
【発明者】  
【住所又は居所】 神奈川県川崎市幸区小向東芝町1番地 株式会社東芝  
研究開発センター内  
【氏名】 吉田 充伸  
【特許出願人】  
【識別番号】 000003078  
【氏名又は名称】 株式会社 東芝  
【代理人】  
【識別番号】 100083161  
【弁理士】  
【氏名又は名称】 外川 英明  
【電話番号】 (03)3457-2512  
【手数料の表示】  
【予納台帳番号】 010261  
【納付金額】 21,000円  
【提出物件の目録】  
【物件名】 明細書 1  
【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 機構シミュレーション方法および機構シミュレーションプログ  
ラム

【特許請求の範囲】

【請求項 1】 連続系方程式を用いて機構の挙動を時間軸に沿ってシミュレーションするダイナミクスシミュレーションと、3次元機構モデルを用いて機構の3次元空間内での幾何学的な動作をシミュレーションするキネマティックスシミュレーションとを併用する機構シミュレーション方法であって、

前記連続系方程式における変数と前記3次元機構モデルの機構要素との対応関係を読み込むステップと、

前記ダイナミクスシミュレーションを実行し前記連続系方程式の変数の値を求めるステップと、

前記対応関係を参照し、前記変数の値に基づいて前記3次元機構モデルの機構要素に対する前記キネマティックスシミュレーションを実行するステップとを具備することを特徴とする機構シミュレーション方法。

【請求項 2】 前記対応関係に基づいて、変数・機構要素対応テーブルを作成するステップをさらに備え、前記キネマティックスシミュレーションは、前記変数・機構要素対応テーブルを参照して行なわれることを特徴とする請求項1記載の機構シミュレーション方法。

【請求項 3】 前記機構要素は、アクチュエータの回転角もしくは移動量を含むことを特徴とする請求項1又は2のいずれか一項に記載の機構シミュレーション方法。

【請求項 4】 前記ダイナミクスシミュレーションは、ハイブリッドモデルを用いたハイブリッドシミュレーションを実行するものであることを特徴とする請求項1乃至3のいずれか一項に記載の機構シミュレーション方法。

【請求項 5】 前記ダイナミクスシミュレーションは、前記機構を制御する機構制御ソフトウェアを含む外部からの制御信号に基づいて動作することを特徴とする請求項1乃至4のいずれか一項に記載の機構シミュレーション方法。

【請求項 6】 連続系方程式を用いて機構の挙動を時間軸に沿ってシミュレーシ

ヨンするダイナミクスシミュレーションと、3次元機構モデルを用いて機構の3次元空間内での幾何学的な動作をシミュレーションするキネマティックスシミュレーションとを併用する手順をコンピュータに実行させる機構シミュレーション方法であって、

前記連続系方程式における変数と前記3次元機構モデルの機構要素との対応関係を読み込む手順と、

前記ダイナミクスシミュレーションを実行し前記連続系方程式の変数の値を求める手順と、

前記対応関係を参照し、前記変数の値に基づいて前記3次元機構モデルの機構要素に対する前記キネマティックスシミュレーションを実行する手順と  
をコンピュータに実行させることを特徴とする機構シミュレーションプログラム。  
。

【請求項7】 前記対応関係に基づいて、変数・機構要素対応テーブルを作成する手順をさらに備え、前記キネマティックスシミュレーションは、前記変数・機構要素対応テーブルを参照して行なわれることを特徴とする請求項6記載の機構シミュレーションプログラム。

【請求項8】 前記ダイナミクスシミュレーションは、ハイブリッドモデルを用いたハイブリッドシミュレーションを実行するものであることを特徴とする請求項6又は7のいずれか一項に記載の機構シミュレーションプログラム。

【請求項9】 前記ダイナミクスシミュレーションは、前記機構を制御する機構制御ソフトウェアを含む外部からの制御信号に基づいて動作することを特徴とする請求項6乃至8のいずれか一項に記載の機構シミュレーションプログラム。

#### 【発明の詳細な説明】

##### 【0001】

##### 【発明の属する技術分野】

本発明は、コンピュータを用いて機構の挙動をシミュレーションする方法およびプログラムに関し、特に微分方程式ないしは代数方程式からなる連続系方程式を用いた機構の時間軸に沿った挙動に関するダイナミクスシミュレーションと、3次元の形状データを含む3次元機構モデルを用いたキネマティックスシミュレ

ーションとを併用するものである。

### 【0002】

#### 【従来の技術】

現在、コンピュータを用いて機械やプラント等の連続系の時間軸上での挙動のシミュレーションを行う際には、対象を微分方程式などでモデル化し、数値積分などの手法を用いて解を求めることが行われている。さらに、複雑な系の挙動を表現するためには、ハイブリッドモデリングと呼ばれる手法が使われることもある。ハイブリッドモデルを用いたシミュレーションは、通常の連立微分方程式によるシミュレーションをさらに高度化したものであり、「ハイブリッドシミュレーション」と呼ばれている。このようなシミュレーション挙動をするシステムを「ハイブリッドシステム」と呼ぶこともある。シミュレーションの目的で作成されるハイブリッドモデルは、概念的には常微分方程式や代数方程式を連立させた連立方程式によって表現される連続系モデルと、イベント発生に伴う状態遷移を表現するための状態遷移モデルとを組み合わせたモデルである。ハイブリッドモデルによれば、連続系モデルで表現される状態が外部からのイベントなどにより瞬時に切り替わるシステムを表現することができ、より高度なモデル表現が可能である。

### 【0003】

ハイブリッドモデルを記述するための言語として、米国ゼロックス社（商標）のパロアルト研究所にて創作されたHCC (Hybrid Concurrent Constraint Programming)と呼ばれる言語がある（下記非特許文献1参照）。HCCは発展途上にあり、現在も米国NASAのエイムズ研究所において研究が進められている。HCCは制約処理プログラミング（コンストレイントプログラミング）と呼ばれる技術の一種であり、連続系モデルを表現する常微分方程式や代数方程式を制約として扱い、これら方程式をそのまま順不同で記述することができる。このような制約記述に、状態遷移を制御する記述を付加してHCC言語のハイブリッドモデルは完成される。HCCによれば、方程式をそのまま制約として羅列する（プログラミングする）ことができ、複雑なモデルを記述することが可能である。

### 【0004】

このようにハイブリッドモデルの技術を用いれば、系の特性を常微分方程式などでモデル表現し、初期状態から時間の推移に従ってどのような挙動を示すかをシミュレーションすることができる。このような微分方程式などの方程式によるモデル化は外乱による過渡応答や振動など動的な挙動を表現できるため、ダイナミクスシミュレーション（動力学シミュレーション）とも呼ばれる。

#### 【0005】

微分方程式などで表現可能な対象や現象を的確にモデル化できるハイブリッドモデルの技術の応用例として、ソフトウェアにより機構が制御されるメカトロニクス機器の機構シミュレーションがある。かかる機構シミュレーションによれば、機構の実機が存在しない状況においても、当該機構を制御する制御ソフトウェアのプロトタイピング、テスト、あるいはデバッグなどが行えるようになる。

#### 【0006】

一方、機構の3次元的な形状データや、部品同志の関節などによる接続関係の情報などを用いて、機構の3次元空間内での幾何学的な動作のシミュレーションを行う技術が知られている。3次元CADの技術や機構シミュレーションソフトウェアがこれに相当し、例えば、特開2001-222572公報（下記特許文献1）に記載の技術も参考にすることができる。このようなソフトウェアは振動特性のような時間軸に沿った系の挙動のシミュレーションはできないが、複雑な位置関係などを扱うことができ、キネマティクスシミュレーション（機構学シミュレーション）と呼ばれることもある。

#### 【0007】

特開2001-222572に記載の技術は、ソフトウェアにより機構が制御されるメカトロニクス機器の機構シミュレーションを、キネマティクスシミュレーションおよび限定的かつすでにパッケージソフトウェアに組込み済みの限られたダイナミクスの範囲において、実現した公知例である。

#### 【0008】

機構シミュレーションソフトウェアの中には、ダイナミクスシミュレーションの機能を取り入れ、機構学と動力学の両方を扱えるようにしたものも存在する。しかし、このようなソフトウェアは、前記HCCで示したようなプログラミング

言語的に、複雑な系のモデル化を柔軟に行えるダイナミクスシミュレーションには対応していない。このため、ユーザがシミュレーションの目的に応じて最適なモデル化手法を選び、それに基づいて作成したモデル記述を用いてキネマティクスシミュレーションとダイナミクスシミュレーションを並立させて実行することができなかった。さらに、より複雑なハイブリッドモデリング技術を用いる場合には、キネマティクスシミュレーションとダイナミクスシミュレーションを連携させることは困難であった。

### 【0009】

#### 【特許文献1】

特開2001-222572公報

### 【0010】

#### 【非特許文献1】

インターネット<URL：<http://www2.parc.com/spl/projects/mbc/publications.html#cclanguages>>

### 【0011】

#### 【発明が解決しようとする課題】

本発明はかかる事情を考慮してなされたものであり、微分方程式などの方程式記述によるダイナミクスモデルやハイブリッドモデルを用いた複雑なダイナミクスモデルと3次元形状データを含むキネマティクスモデルを容易に連携させることを可能とすることを目的とする。

### 【0012】

また、これにより、ダイナミクスシミュレーションとキネマティクスシミュレーションを並立した高度なシミュレーションを、ソフトウェアにより機構が制御されるメカトロニクス機器の実機レスシミュレーションに適用可能にすることを目的とする。

### 【0013】

#### 【課題を解決するための手段】

上記課題を解決し目的を達成するために、本発明では、連続系方程式を用いて機構の挙動を時間軸に沿ってシミュレーションするダイナミクスシミュレーション

と、3次元機構モデルを用いて機構の3次元空間内での幾何学的な動作をシミュレーションするキネマティックスシミュレーションと併用する機構シミュレーション方法であって、前記連続系方程式における変数と前記3次元機構モデルの機構要素との対応関係を読み込むステップと、前記ダイナミクスシミュレーションを実行し前記連続系方程式の変数の値を求めるステップと、前記対応関係を参照し、前記変数の値に基づいて前記3次元機構モデルの機構要素に対する前記キネマティックスシミュレーションを実行するステップとを具備することを特徴とする機構シミュレーション方法を提供する。

#### 【0014】

また、本発明では、連続系方程式を用いて機構の挙動を時間軸に沿ってシミュレーションするダイナミクスシミュレーションと、3次元機構モデルを用いて機構の3次元空間内での幾何学的な動作をシミュレーションするキネマティックスシミュレーションと併用する手順をコンピュータに実行させる機構シミュレーション方法であって、前記連続系方程式における変数と前記3次元機構モデルの機構要素との対応関係を読み込む手順と、前記ダイナミクスシミュレーションを実行し前記連続系方程式の変数の値を求める手順と、前記対応関係を参照し、前記変数の値に基づいて前記3次元機構モデルの機構要素に対する前記キネマティックスシミュレーションを実行する手順とをコンピュータに実行させることを特徴とする機構シミュレーションプログラムを併せて提供する。

#### 【0015】

##### 【発明の実施の形態】

以下、図面を参照しながら本発明の実施形態を説明する。なお、以下では、ダイナミクスシミュレータとしてハイブリッドモデリング技術を採用し、かつ、機構制御ソフトウェアないしは機構制御ソフトウェアのシミュレータと連携して動作する最も複雑な実施形態の1つの態様をもとに説明する。

#### 【0016】

図1は、本発明に係る機構シミュレータの概略構成を示すブロック図であり、図2は、かかる機構シミュレータが機構制御ソフトウェアないしは機構制御ソフトウェアのシミュレータと連携する場合の構成を示している。

### 【0017】

機構シミュレータはダイナミクスシミュレーション部101およびキネマティクスシミュレーション部103および変数・機構要素対応テーブル105から構成され、機構制御ソフトウェアないしは機構制御ソフトウェアのシミュレータ108と制御信号109をやり取りする。なお、本実施形態は一般的なコンピュータ107を用いて構成することができ、その基本的なハードウェア構成として、図示しないが中央演算ユニット（CPU）、メモリ、外部記録装置、通信インターフェース（I/F）、および表示装置やキーボード、マウス等の入力装置を備える。また、これらのハードウェアを制御するためのオペレーティングシステム（OS）を備える。また、本発明の実施形態に係る機構シミュレータは、このようなオペレーティングシステム上で動作するアプリケーションソフトウェアとして実装することができる。

### 【0018】

シミュレーション対象である機構の動力学的なモデル記述は、後で詳細に説明するハイブリッドモデル記述言語により記述され、ファイル（ハイブリッドモデル記述）102としてあらかじめ記憶されている。一方、3次元形状情報を含む機構モデルデータも所定のフォーマットに従いファイル（3次元機構モデル）104としてあらかじめ保存されている。これらのデータはシミュレーション開始時に読み込まれる。この段階では、ダイナミクスシミュレーション部101およびキネマティクスシミュレーション部103がそれぞれ単独に動作することが可能である。次に、あらかじめ保存されていたダイナミクスモデル記述に現れる変数と3次元機構モデルの機構要素との対応関係データ106を読み込む、ないしはPCのキーボードなど入力デバイスより直接データを入力することにより、変数・機構要素対応テーブル105のデータがセットされる。これにより、ダイナミクスシミュレーション部101およびキネマティクスシミュレーション部103が連携して動作する準備が整う。

### 【0019】

実際のシミュレーションにおいては、モータなどアクチュエータへの加速コマンドや加速レートのパラメータ設定などの情報が、機構制御ソフトウェアないし

は機構制御ソフトウェアのシミュレータ108から送信され、それに基づきハイブリッドモデルとして記述された適切な状態遷移と方程式に基づく数値積分による系の挙動が計算される。ハイブリッドモデルにより、どのようにダイナミクスが表現され、状態遷移が実現できるかについては、後で詳細に説明する。

### 【0020】

実際の機構制御系においては、機構制御ソフトウェアからのコマンドやパラメータは、一定の時間サイクルごとに指定されたI/Oポートを読み込みにいくPolling処理によって取得される。そのためシミュレーションにおいても、これと同じ時間間隔ごとにシミュレーションをすすめていく。

### 【0021】

すなわち、図3に示されるフローチャートのように、ハイブリッドモデルの読み込みステップ501および3次元機構モデル読み込みステップ502を実行し（501と502の順序は入れ替わった実施例としても全く問題はない）、シミュレーションを開始する。前記一定のサイクルごとに制御信号を読み込みに行く制御信号受信ステップ503を行い、それに基づく適切な状態遷移と方程式に基づく数値積分による系の挙動が計算されるダイナミクスシミュレーションステップ504が実行される。これにより、各変数の値が計算されるので、変数と機構要素の対応関係をチェックしキネマティクスシミュレーション部に値を送信するステップ505が実行され、その値に基づきキネマティクスシミュレーションがステップ506で実行される。機構制御系は、機構からセンサー情報を受け取って、新たな制御コマンドの必要性を判断するので、ダイナミクスシミュレータ101ないしはキネマティクスシミュレータ103から得られる主にセンサーデータを機構制御系に送信するステップ507が実行される。このあと、あらかじめ指定された時間だけシミュレーションが進んだかどうかを判断するステップ508を行い、まだ実行途中であれば次の時間に進めるために、データの記録などの処理を行うステップ509が実行され、ふたたび制御信号の受信ステップ503にもどる。

### 【0022】

このような本実施形態に係るダイナミクスシミュレーションのためのハイブリ

ッドモデル記述102の詳細を、具体例を挙げて説明する。

### 【0023】

図4および図5は、具体例に係るハイブリッドモデルの記述対象である機構を示す図である。この機構は、バルブ301、バネ303、およびピストン302を備えた簡素な構造を有するシリンダ装置である。

### 【0024】

バルブ301は外部からの指令（イベント）に応じて開閉動作する。これによりシリンダ装置内の空気の流れを図2のように右側に変更するイベントを以下、「Right」と呼び、空気の流れを図3のように左側に変更するイベントを「Left」と呼ぶ。図4は、バルブ301にRightのイベントが与えられた状態を示しており、ピストン302には紙面左向きへの力が作用している。この状態を示す運動方程式はシリンダ装置の下部に示してあるように、「 $-F = m \ddot{x}$ 」である。これに対し図5は、バルブ301にLeftのイベントが与えられた状態を示しており、空気の流れの向きが変わり、運動方程式は同図のように「 $F = m \ddot{x}$ 」に変化している。

### 【0025】

図6は、このような状態変化とそれぞれの状態に対応する運動方程式を状態遷移図として表現したものである。ハイブリッドモデルは、この図6に示されるような状態遷移と、各状態の記述が微分方程式や代数方程式で表現されるものを指す。図6によれば、状態が2つあって、かかる2つの状態間に状態遷移が存在することがわかる。

### 【0026】

図7は、図6の状態遷移図をもとに具体的なハイブリッドモデルの内容をHCC (Hybrid Concurrent Constraint Programming)言語で記述したプログラムの一例を示す図である。図7において、（ソース）プログラムの論理行番号を仮にL1～L8とする。L3、L4、およびL8は、この機構の初期状態やバルブ操作タイミングなどの運転条件の記述に相当し、L5およびL6は、図6に示した状態遷移の表現記述である。

### 【0027】

HCCでは、運動方程式は、図から分かるようにプログラム内でそのまま記述することができる。また、それぞれの状態へ遷移する条件は、「always if」に統いて記述し、また、それぞれの状態から遷移していく条件は「watching」に統けて記述すればよい。

### 【0028】

なお、HCCでは、プログラムの記述の順序（図7における論理行番号L1→L8の順序）にそって実行されるわけではない。HCCでは、個別のプログラム記述のうち、シミュレーションを実行する時間軸に沿って成立するものが探索され、実行される。すなわち、論理行番号L1→L8の順序は、実行順序とは関係がない。たとえば、シミュレーションを開始した時点では、L3およびL8のみが有効である。ここで、イベントRight（ev1）がL3により発生するため、L6の前提条件であるRightが有効となり、L6に記述されている運動方程式eq2が有効になる。つまり、図4の左側の状態からシミュレーションが実行されることになる。

### 【0029】

さらに、時間が50になるとL4が有効となり、イベントLeft（ev2）が発生し、L6の遷移条件（「watching」以下、すなわちLeft）が有効となって、L6の運動方程式eq2が無効となる。これに代わって、L5の前提条件が有効となり、運動方程式eq1が有効となる。

### 【0030】

なお、以上のようなプログラム例は外部からのイベント（ev3, ev4）によって状態が遷移する場合を記述したものであったが、勿論、内部の状況によつて状態を変化させてもよい。たとえば、図4においてバルブ301が切り替えられない場合には、移動するピストン302がバネ303に接触し、該バネ303からの反力を受けるようになる。すなわちピストン302の位置に関して、外部からのイベントが無い場合でも状態遷移が起こる場合が存在する。このような場合は、例えばxが正であるかどうかといった内部変数の評価式（不等式）による評価結果に基づいて状態遷移の必要性を判断できる。

### 【0031】

一般的に、ハイブリッドモデルは、常微分方程式や代数方程式をあるいはこれらを連立させた連立方程式（連続系方程式）によって表現される連続系モデルと、イベント発生に伴う状態遷移を表現するための状態遷移モデルとを組み合わせたモデルである。ハイブリッドモデルによれば、連続系モデルで表現される状態が外部からのイベントなどにより瞬時に切り替わるシステムを表現することができる。

#### 【0032】

さらに、図8を用いて、ハイブリッドモデル記述を用いたダイナミクスシミュレーションの詳細について説明する。図8は、図1および図2に示したダイナミクスシミュレーション部101の一実施形態を示したブロック図である。

#### 【0033】

本実施形態は、ハイブリッドモデル前処理部801と、ハイブリッドモデルシミュレーション実行部802とにより構成されている。ハイブリッドモデル記述102は、HCC言語で記述されたソースプログラムであって、本実施形態に係るハイブリッドモデル前処理部801への入力である。制御信号109はハイブリッドモデルシミュレーション実行部802への入力であり、この制御信号109は機構制御ソフトウェアもしくは機構制御ソフトウェアのシミュレータから与えられる。また、本実施形態に係る機構ハイブリッドモデルシミュレーション実行部802からの出力は、シミュレーション結果としての変数値の演算結果およびその時間履歴であり、変数値時間履歴記憶部805に対して出力される。

#### 【0034】

図8に示すように、ハイブリッドモデル前処理部801は、モデル方程式制御情報解析部811を備える。また、ハイブリッドモデルシミュレーション実行部802は、方程式構文解析部812、方程式データ記憶部814、連続系方程式切り替え部815、及び連続系シミュレーション部803とを備える。なお、本実施形態は一般的なコンピュータを用いて構成することができ、その基本的なハードウェア構成として、図示しないが中央演算ユニット（CPU）、メモリ、外部記録装置、通信インターフェース（I/F）、および表示装置やキーボード、マウス等の入力装置を備える。また、これらのハードウェアを制御するためのオ

ペレーティングシステム（OS）を備える。また、本発明の実施形態に係る機構シミュレータは、このようなオペレーティングシステム上で動作するアプリケーションソフトウェアとして実装することができる。

### 【0035】

次に、ハイブリッドモデル前処理部801における処理について説明する。ハイブリッドモデル記述102は、まずハイブリッドモデル前処理部801のモデル方程式制御情報解析部811において処理され、モデル方程式登録プログラム806、モデル方程式制御プログラム807が生成される。また、ハイブリッドモデルシミュレーション実行部102を構成するソフトウェアモジュールとして、モデル方程式の登録を行うための関数及び連続系方程式を切り替えるための関数がA P I（Application Program Interface）関数として提供される。モデル方程式登録プログラム806およびモデル方程式制御プログラム807は、該当する上記A P I関数を呼び出す記述を、入力されたハイブリッドモデル記述102に沿って適切に組み合わせたプログラムである。この観点から考えると、ハイブリッドモデル前処理部801は、入力をハイブリッドモデル記述102とし、出力を例えばC言語のA P I関数呼び出しの記述を含むCプログラム（ソース）とするような、一種のコンパイラと考えることもできる。このようなモデル方程式登録プログラム806とモデル方程式制御プログラム807は、さらにC言語などのコンパイラによりコンパイルされ、例えば実行時に動的にリンク可能なライブラリが生成される。ハイブリッドモデルシミュレーション実行部802は、シミュレーション実行にあたって、生成された動的リンクライブラリがリンクされ、入力ハイブリッドモデルを忠実に再現するシミュレーションプログラムが完成し、実行可能になる。その実行時には、まず方程式構文解析部812を起動するA P I関数が呼ばれ、その後に連続系方程式切り替えのA P I関数群が実行されて連続系シミュレーションが遂行される。

### 【0036】

ハイブリッドモデルシミュレーション部802のアプリケーションインターフェースを構成する具体的なソフトウェアモジュールの仕様などは様々考えられるが、ここでは説明の都合上、以下の3つのA P I関数が最低定義されているとす

る。なお、プログラミング言語はC言語とする。

### 【0037】

#### 【数1】

```
int XXX_AddEqnData(char *eqn, int *err)
int XXX_ActivateEqn(int eqnid)
int XXX_DeActivateEqn(int eqnid)
```

### 【0038】

1つ目のA P I 関数XXX\_AddEqnDataは、1つの連続系方程式を表す文字列のポインタを引数に指定する。XXX\_AddEqnDataは、この連続系方程式を構文解析し、連続系方程式の記述をシミュレーション実行可能なデータ構造（内部データ表現）に変換し、かかる内部データ表現を方程式データ記憶部814に登録する処理を行う。なお、ここの連続系方程式には、ユニークなID番号が割り当てられる。

### 【0039】

たとえば「 $ab/\cos(a-(c+b))-3c$ 」という式が与えられたと仮定すると、上記内部データ表現として図9のような木構造を生成する。この木構造において、例えば参照数字61は線形多項式の親ノード（節）、62は掛け算のノード、63は割り算のノード、64は外部関数（四則演算以外の意）のノード、65は線形多項式を構成する各項のノードを表している。本例において、木構造の葉に相当するものはすべて変数(a,b,c)であり、これらに実数の係数が加わって線形式となる。線形式はcosなどの外部関数の引数になったり、掛け算や割り算の対象となる。変数には、別途、値が確定しているかどうかのフラグが設けられており、またこのような木構造のデータに基づいて該変数の現在の値が保持される。木構造のすべての葉の値（すなわち変数の値）が確定していれば、式の値を計算することができる。方程式データ記憶部114では、式の値の計算などを高速に行うことができるよう、予め内部のデータ構造をつなぎ合わせて木構造を構成している。

### 【0040】

上記処理において何らかのエラーが発生した場合には、errにエラーコードが

セットされる。正常に処理が終了した場合は、登録された方程式のID番号を返り値とする。

#### 【0041】

2番目のAPI関数XXX\_ActivateEqnは、引数に指定された方程式のID番号に相当する方程式を有効にする。もし、すでに有効となっている方程式が指定されている場合には何もしない。返り値はエラーコードである。

#### 【0042】

3番目のAPI関数XXX\_DeActivateEqnは、XXX\_ActivateEqnとは逆に、引数に指定された方程式のID番号に相当する方程式を無効にする。すでに無効となっている方程式が指定された場合には何もしない。

#### 【0043】

モデル方程式制御情報解析部811は、まずXXX\_AddEqnDataを必要な方程式について順に呼ぶ関数(InitEqnData)を生成する。これがモデル方程式登録プログラム806に相当する。

#### 【0044】

また、モデル方程式制御情報解析部811は、シミュレーション実行の際に、時間が $\Delta t$ 進むごとに条件のチェックおよび方程式の変更(入れ替え)を行う関数(Chan

geEqn)も生成する。これはモデル方程式制御プログラム807に相当する。

上記したようなハイブリッドモデル前処理部801における処理により、例えば、図7に示したハイブリッドモデル記述について、以下のようなC言語のソースプログラムが自動生成される。

#### 【0045】

## 【数2】

```

static char eqn1[] = "f = m x";
static char eqn2[] = "-f = m x";
static int eqn1id;
static int eqn2id;
int InitEqnData()
{
    int err;
    eqn1id = XXX_AddEqnData(eqn1,&err);
    if(err != 0) return err;
    eqn2id = XXX_AddEqnData(eqn2,&err);
    if(err != 0) return err;
    return 0;
}
int ChangeEqn()
{
    int err;
    BOOL GetEvent(char *eventname);
    if( GetEvent(Left) ){
        err = XXX_ActivateEqn(eqn1id);
        if( err != 0 ) return err;
        XXX_DeActivateEqn(eqn2id);
        if( err != 0 ) return err;
    }
    if( GetEvent(Right) ){
        XXX_ActivateEqn(eqn2id);
        if( err != 0 ) return err;
        XXX_DeActivateEqn(eqn1id);
        if( err != 0 ) return err;
    }
}

```

## 【0046】

なお、GetEventは、名前で指定されたイベントが生起しているかどうかをチェックする関数である。

以上のプログラムは、上述したようにC言語コンパイラによってコンパイルされ、さらに動的リンクライブラリの形式に整えられ、実行時にリンクされる。

## 【0047】

なお、本実施形態では、プログラム言語としてC言語を用いた例について説明したが、本発明はこれに限定されるものではなく、例えばCPP言語、Spec-C言語等の他のプログラム言語を用いててもよい。

#### 【0048】

次に、シミュレーションの実行について説明する。シミュレーション実行時においては、ハイブリッドモデルシミュレーション実行部802が起動され、図2に示す機構制御ソフトウェアシミュレータ108などから得られる制御信号109を受信しながら、連続系方程式の値を計算することでシミュレーション実行が行われる。このとき、連続系方程式切り替え部815は、上記したモデル方程式制御プログラム807に基づいて、連続系方程式の切り替えを有効・無効のフラグを用いて実行する。図4の状態では、図7の運動方程式eq1は無効であり、運動方程式eq2が有効になっている。ここで、Leftのイベントが発生した図3の状況においては、図7の運動方程式eq1を有効にし、運動方程式eq2を無効にするようフラグを操作する。これら有効・無効のフラグは方程式データ記憶部814に記憶される方程式それぞれの属性データとして管理される。

#### 【0049】

連続系シミュレーション部803は、方程式データ記憶部814を参照し、同記憶部814に木構造の形式で格納されている連続系方程式の内部データ表現を演算対象として、時間ステップづつ数値積分を実行する。シミュレーションは、常微分方程式及び代数多項式の連立からなる非線形連立方程式についての初期値問題である。このため、例えば図4に示される初期状態が与えられている。具体的には、例えば一般によく使われているルンゲクッタアルゴリズムを用いて変数の値を計算する。

#### 【0050】

必要なデータは機構シミュレータから出力を行い、さらに連続系方程式切り替え部815の処理に戻り、上記の処理を繰り返すことにより必要な時間のシミュレーションを実行する。シミュレーション結果は、変数値時間履歴記憶部805に保存され、シミュレーション終了後の分析などに利用される。

#### 【0051】

次に、図10から図12を用いて、3次元機構モデル104およびキネマティクスシミュレーション部103の具体的な実施形態を説明する。

#### 【0052】

3次元機構モデル104では、図10に示すように個々の部品の形状が、その構成要素である面や稜線、頂点などの要素の集合として表現される。これら面や稜線、頂点などは部品の形状の境界を表現しているため、境界表現の形状モデルと呼ばれ、3次元のCADシステムなどで広く用いられている。このように表現された個々の部品の関係は、さらに部品の部分形状間の関係として表現される。この例では、部品401、部品402、部品403の3次元形状があり、部品401の部分形状である平面404と部品402の部分形状である平面406が一致し、同様に円筒面405と円筒面407が同軸であるという関係が定義されている。同様に部品402と部品403については、部分形状である平面408と平面410の一致関係、および平面409と平面411との一致関係が定義されている。

#### 【0053】

図11は、図10で定義された関係が計算機メモリなどの記憶手段にどのような形式で記憶されるかをグラフの形式で表現したものである。3次元形状の情報および部分形状の情報の他に、平面同士の一致関係412、414、415および円筒同士の同軸関係413がデータとして存在する。このような情報から部品401、402、403の相対的位置関係を計算する機能は幾何拘束処理ライブラリとして既存のソフトウェアが提供されている。具体的には、それぞれの部品には、部品特有のローカル座標系が設定されており、この座標系と空間内に固定されたワールド座標系との変換マトリックスという形でこの位置関係が表現される。すなわち、図11に示されている一致などの関係から、部品401と部品402、部品403の位置を表現する変換マトリックスが幾何拘束処理ライブラリなどにより自動的に計算される。

#### 【0054】

図12は、部分形状の間で定義された一致関係、すなわち部分形状間の拘束関係をすべて満足するように算出された部品間の位置関係の例を示す。すなわち、

幾何拘束処理ライブラリが自動算出した部品の変換マトリックスを部品の形状データに作用させて得られた部品の位置を示している。このように組み立てられた部品には、416で示される回転自由度と417で示される並進自由度が存在する。実際に機構として作用させるためには、これらにモータなどのアクチュエータを装着し、外部の信号に基づいて駆動することが行われる。そこで、回転自由度416に装着されるアクチュエータにJoint1という名前を付け、並進自由度417に装着されるアクチュエータにSlide1という名前を付ける。キネマティクスシミュレーション部103はアクチュエータの名前とその値を指定すると、図12のような3次元グラフィックスのアニメーションにて機構の動きを確認することなどが可能になる。また、図12に示されている距離418を計算する機能などはキネマティクスシミュレーション部103に実装されており、部品403が溝の端まで到達したかどうかを検知するリミットスイッチセンサーの機能をこの距離の値の判定によってシミュレートすることができる。このセンサーの名前をSwitch1とすると、キネマティクスシミュレーション部103はSwitch1という名前でセンサー状態を問い合わせると、そのときのセンサーの状態を得ることができる。

#### 【0055】

今、アクチュエータSlide1は、実際には図4のシリング装置に取り付けられているとする。この場合、図7のダイナミクスシミュレータの記述における変数xが機構要素であるアクチュエータSlide1に対応する。さらにJoint1に相当するモータの挙動がダイナミクスシミュレータの記述において変数yとなっているとすると、変数・機構要素対応テーブル105は具体的に図13に示すようになる。

#### 【0056】

このような例においては、図3のフローチャートにおいて、図7のダイナミクスモデル記述に基づきステップ504においてxおよびyの値の変化が計算される。次にステップ505において、変数・機構要素対応テーブル105である図12のテーブルが参照され、キネマティクスシミュレーション部103に対して、xの値とSlide1という名前の組でデータが送信され、同じくyの値とJoint1と

いう名前の組でデータが送信される。ステップ506では、図12のような個々の部品の3次元的な位置が計算され、それに基づき距離418も算出される。距離418の値によりセンサーSwitch1の値もセットされる。ステップ507では、Switch1の値などが機構制御系へ送信される。

### 【0057】

なお、本発明は上述した実施形態に限定されず種々変形して実施可能である。たとえば、ダイナミクスシミュレーションはハイブリッドモデルの例で説明したが、状態遷移を伴わない場合には、ハイブリッドモデルを利用しないでもダイナミクスシミュレーション部101を構成することが可能である。

### 【0058】

#### 【発明の効果】

以上説明したように、本発明によれば、ハイブリッドモデルを用いるようなダイナミクスシミュレーションと3次元キネマティクスシミュレーションを簡便かつ効果的に連携させることが可能となり、かつダイナミクスシミュレーション部は、ハイブリッドモデル記述言語のようなプログラミング言語により自由に記述することが可能となる。これにより、複雑な機構系を簡便かつ正確にモデル化でき、該機構系を制御する制御ソフトウェアとの連携シミュレーションにも好適なシミュレーション方法およびプログラムを提供できる。

#### 【図面の簡単な説明】

【図1】 本発明にかかる機構シミュレータの概略構成を示すブロック図。

【図2】 本発明にかかる機構シミュレータと機構制御ソフトウェアの連携した状態を示すブロック図。

【図3】 機構シミュレーションの処理手順を示すフローチャート。

【図4】 ハイブリッドモデル記述を説明するための具体例に係るシリンダ装置のある状態を示す図。

【図5】 ハイブリッドモデル記述を説明するための具体例に係るシリンダ装置の別の状態を示す図。

【図6】 ハイブリッドモデル記述を説明するための具体例に係るシリンダ装置の状態遷移を示す図。

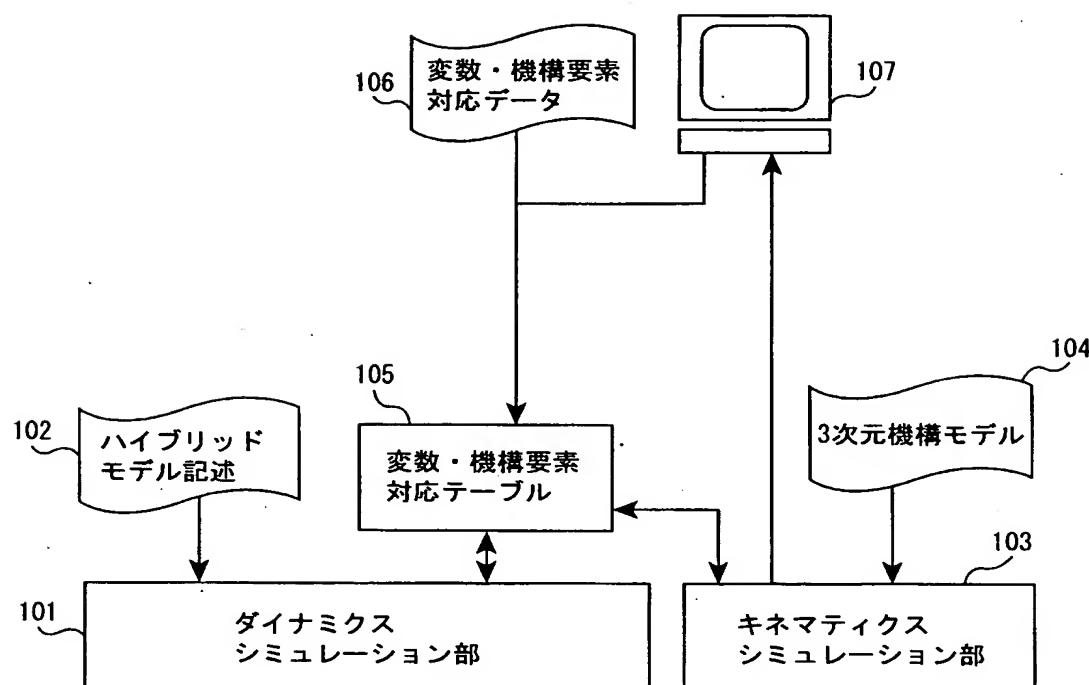
- 【図7】 ハイブリッドモデル記述の内容を示す図。
- 【図8】 ダイナミクスシミュレーション部の一実施形態を示しすブロック図。
- 【図9】 1つの連続系方程式を構文解析した結果得られる内部データ表現の説明図。
- 【図10】 3次元機構モデルにおける構成要素に関する説明図。
- 【図11】 3次元機構モデルで定義された関係が計算機メモリなどの記憶手段にどのような形式で記憶されるかをグラフの形式で表現した図。
- 【図12】 機構の動きを3次元グラフィックスのアニメーションにて表現した図。
- 【図13】 変数・機構要素対応テーブル説明図。

【符号の説明】

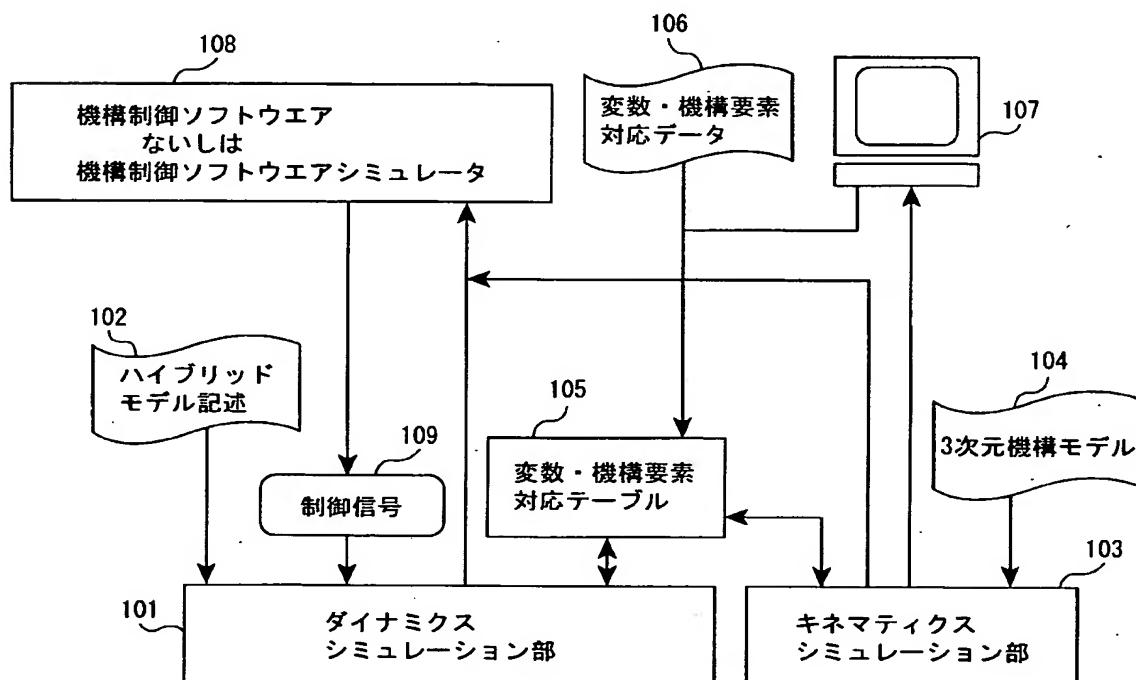
- 101 ダイナミクスシミュレーション部  
102 ハイブリッドモデル記述  
103 キネマティクスシミュレーション部  
104 3次元機構モデル  
105 変数・機構要素対応テーブル  
106 変数・機構要素対応データ  
107 コンピュータ  
109 制御信号

【書類名】 図面

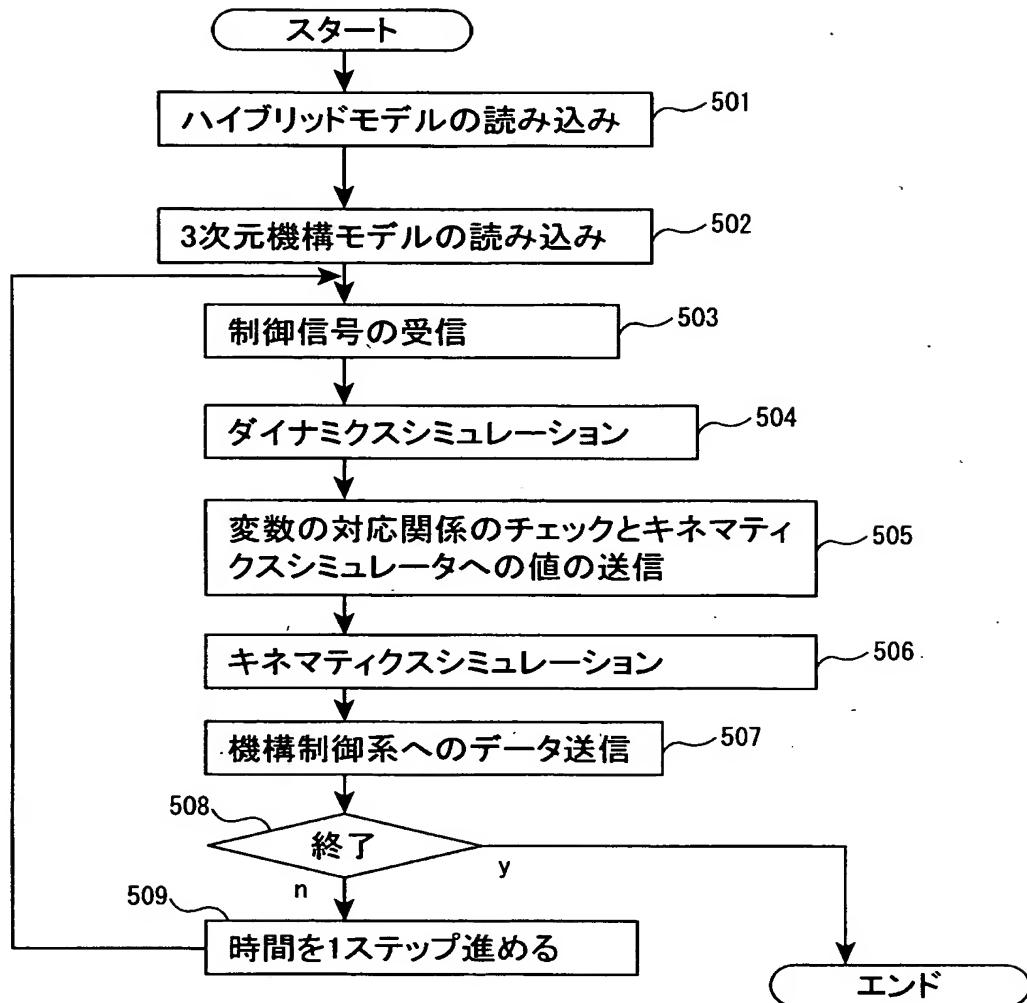
【図1】



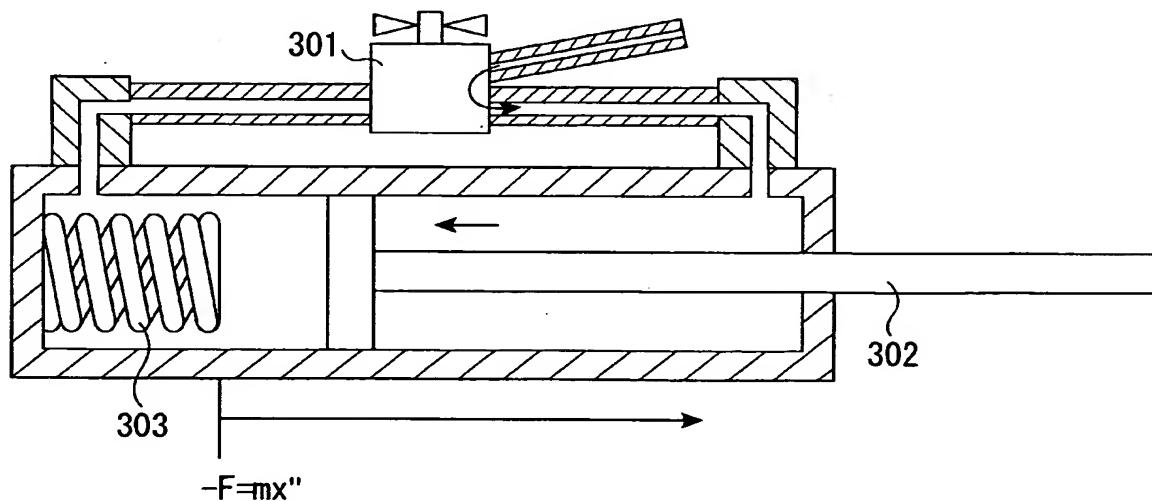
【図2】



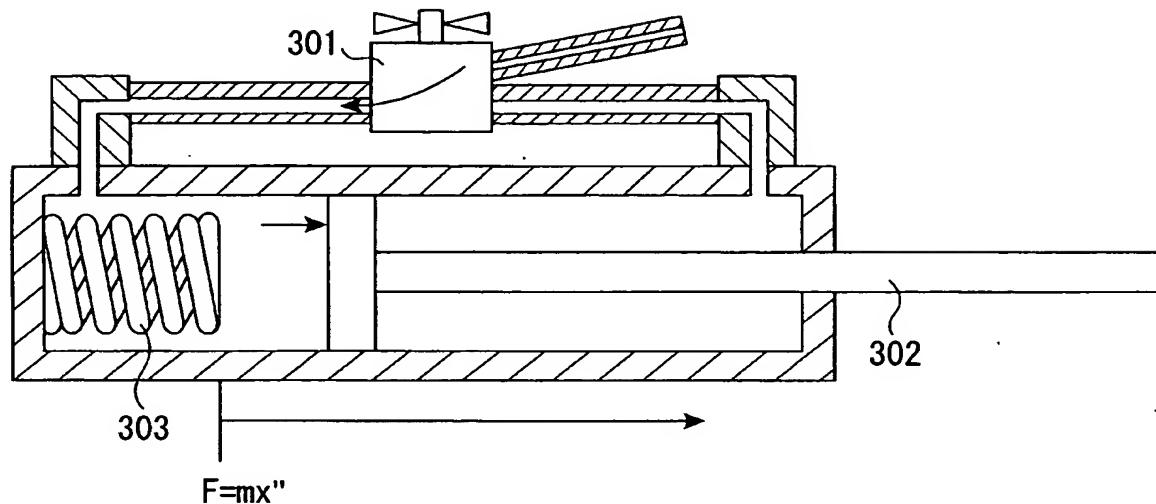
【図3】



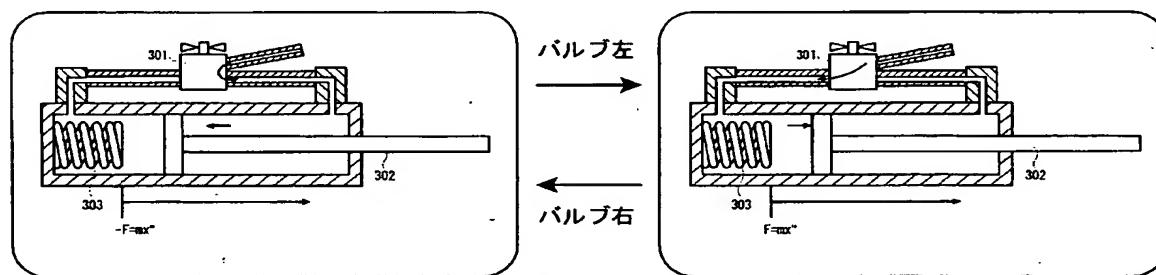
【図4】



【図5】



【図6】



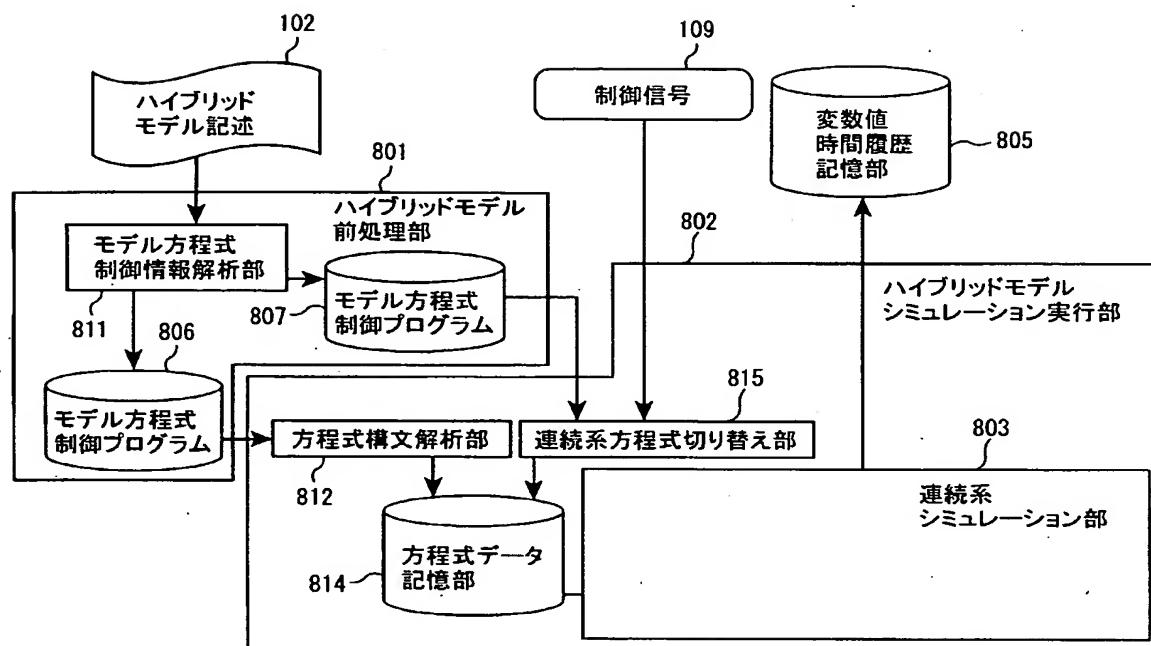
【図7】

```

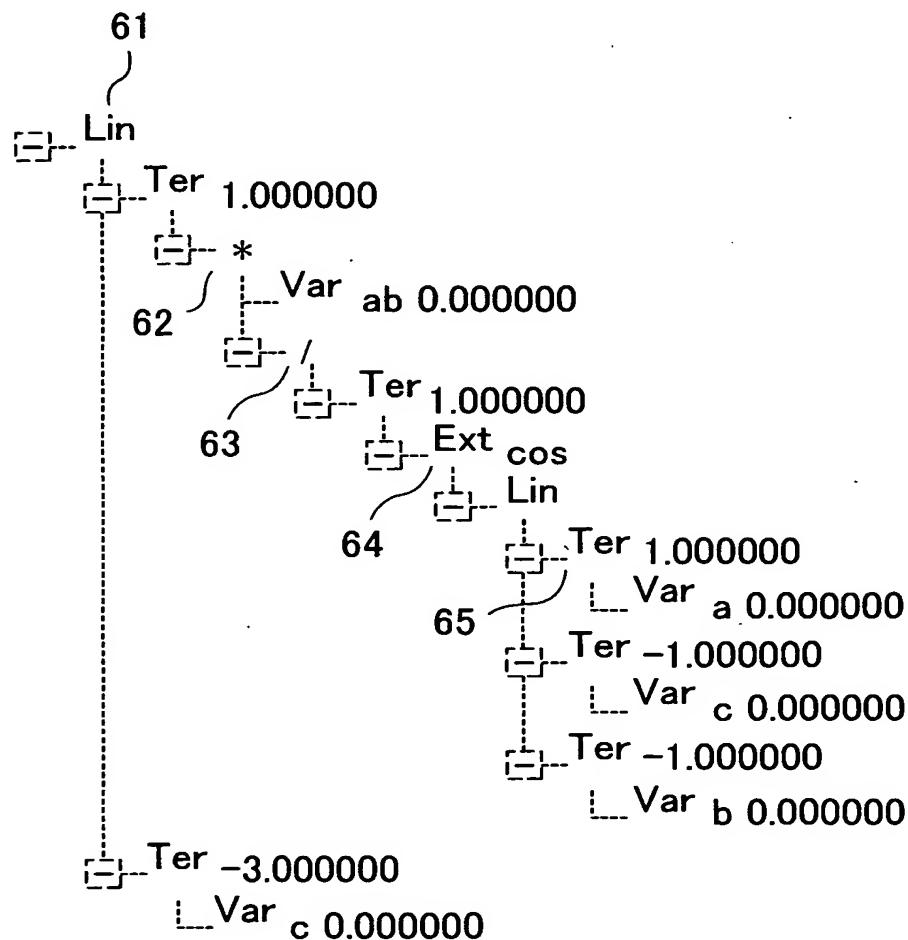
L1 #define m 1
L2 #define f 100
L3 Right ~ ev1
L4 wait 50 do Left ~ ev3
L5 always if Left then do always [F = m * x''] watching Right,
L6 always if Right then do always [-F = m * x''] watching Left,
L7 sample(x),
L8 x = 0, x' = 0,
    
```

The code consists of several lines (L1-L8) defining constants and performing actions. Lines L3-L8 involve events ev1 through ev4 and equations eq1 and eq2. The code uses a block-based programming language syntax where actions are triggered by events and monitored by equations.

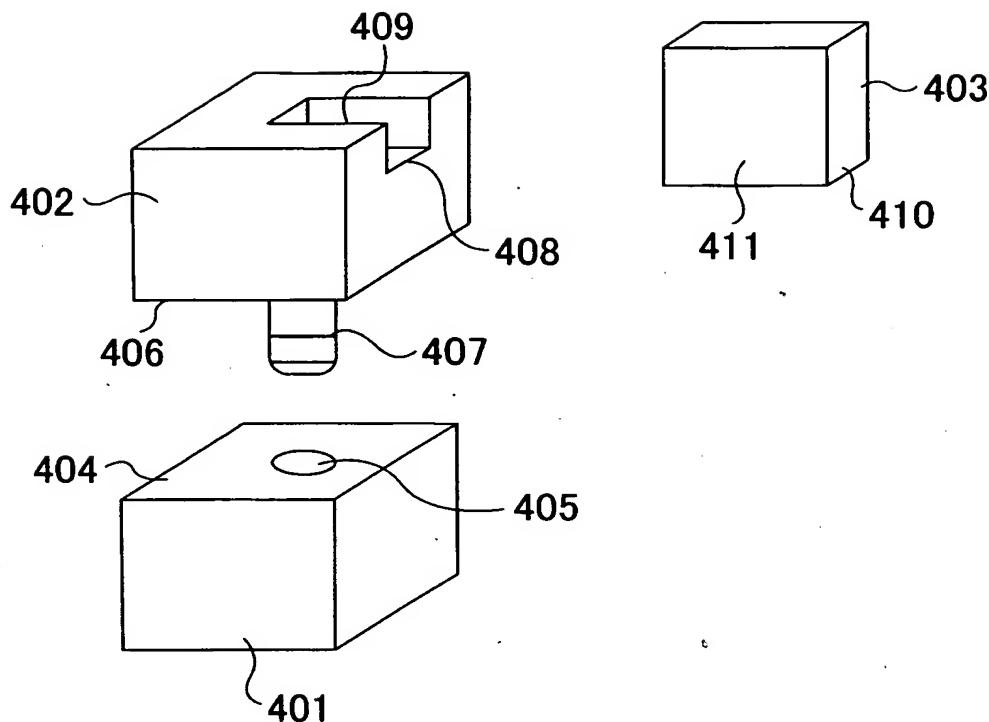
【図 8】



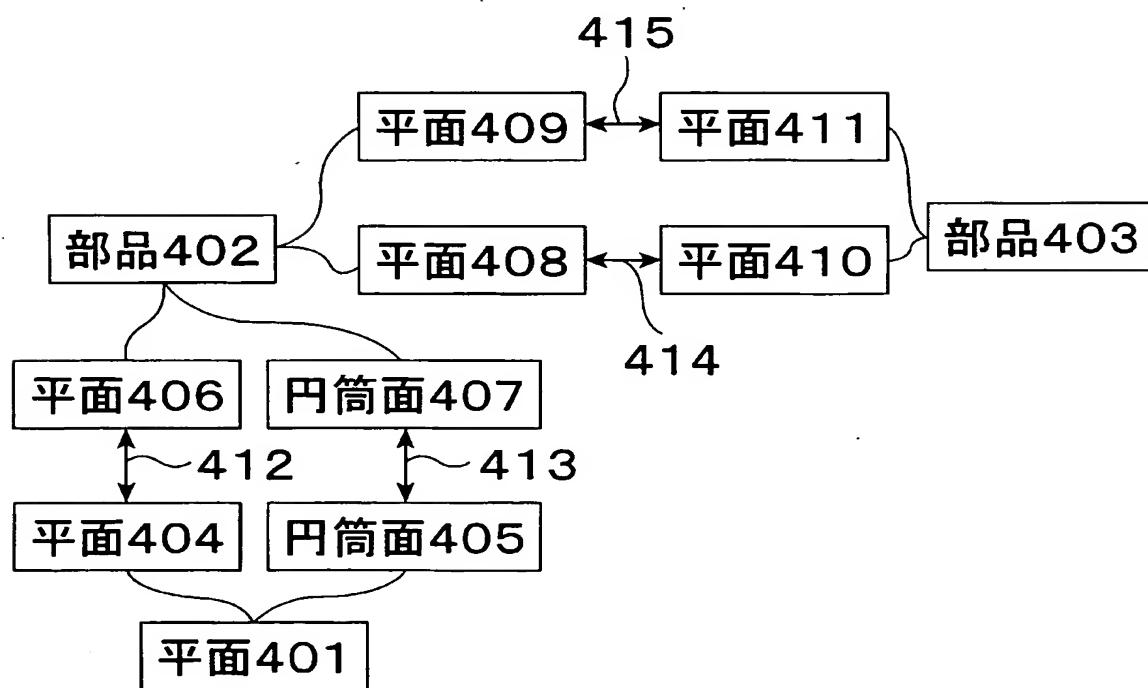
【図 9】



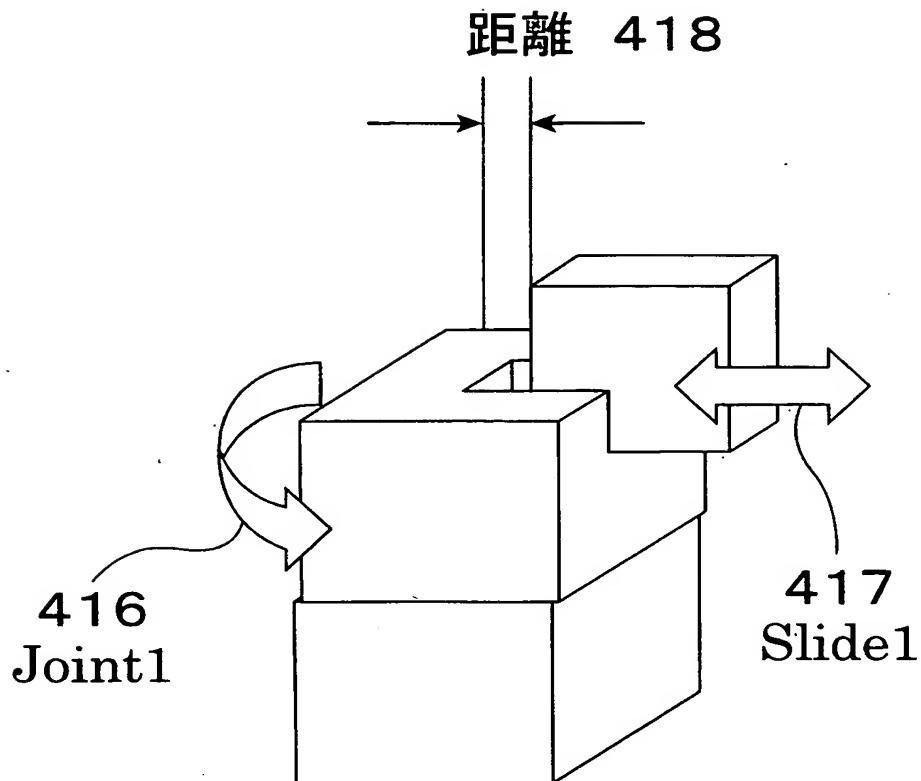
【図10】



【図11】



【図12】



【図13】

変数	機構要素名
x	Slide1
y	Joint1

【書類名】 要約書

【要約】

【課題】 連続系方程式を用いて機構の挙動を時間軸に沿ってシミュレーションするダイナミクスシミュレーションと、3次元機構モデルを用いたキネマティックスシミュレーションとを併用する機構シミュレーション方法およびプログラムの提供。

【解決手段】 ダイナミクスシミュレーションと、キネマティックスシミュレーションとを併用する機構シミュレーション方法及びであって、連続系方程式における変数と3次元機構モデルの機構要素との対応関係を読み込むステップ501、502と、ダイナミクスシミュレーションを実行し連続系方程式の変数の値を求めるステップ504と、対応関係を参照し、変数の値に基づいて3次元機構モデルの機構要素に対するキネマティックスシミュレーションを実行するステップ506とを具備することを特徴とする機構シミュレーション方法及びプログラム。

【選択図】 図3

**認定・付加情報**

特許出願の番号 特願2002-376209  
受付番号 50201970898  
書類名 特許願  
担当官 第七担当上席 0096  
作成日 平成15年 1月 6日

## &lt;認定情報・付加情報&gt;

【提出日】 平成14年12月26日

次頁無

特願2002-376209

出願人履歴情報

識別番号 [000003078]

1. 変更年月日 2001年 7月 2日

[変更理由] 住所変更

住 所 東京都港区芝浦一丁目1番1号  
氏 名 株式会社東芝

2. 変更年月日 2003年 5月 9日

[変更理由] 名称変更

住 所 東京都港区芝浦一丁目1番1号  
氏 名 株式会社東芝